AD-A285 175

TASK: UU04
CDRL: 05203
24 February 1993

Ada PCTE Binding (AdaPCTE) Version 0.3 SunOS Implementation

Informal Technical Data

DTIC
SEP 30 1994

94-31244

STARS-UC-05203/004/00
24 February 1993

DTIC QUALITY INSPECTED 3

9 1

TASK: UU04
CDRL: 05203
24 February 1993

VERSION DESCRIPTION DOCUMENT

For

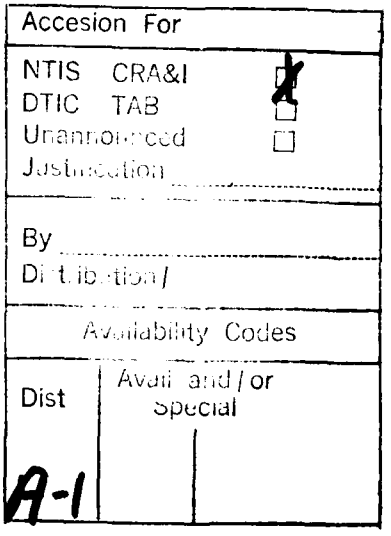
SOFTWARE TECHNOLOGY FOR ADAPTABLE, RELIABLE SYSTEMS (STARS)

Ada PCTE Binding (AdaPCTE)
Version 0.3
SunOS Implementation

STARS-UC-05203/004/00
24 February 1993

Data Type: A005, Informal Technical Data

CONTRACT NO. F19628-88-D-0031
Delivery Order 0011



Prepared for:

Electronic Systems Center
Air Force Systems Command, USAF
Hanscom AFB, MA 01731-5000

Prepared by:

Paramax Systems Corporation
12010 Sunrise Valley Drive
Reston, VA 22091

Distribution Statement "A"
per DoD Directive 5230.24
Authorized for public release; Distribution is unlimited.

TASK: UU04
CDRL: 05203
24 February 1993

Data ID: STARS-UC-05203/004/00

Distribution Statement "A"
per DoD Directive 5230.24
Authorized for public release; Distribution is unlimited.

Copyright 1993, Paramax Systems Corporation, Reston, Virginia
Copyright is assigned to the U.S. Government, upon delivery thereto, in accordance with the DFAR Special Works Clause.

Developed by: Paramax Systems Corporation

This software, developed under the Software Technology for Adaptable, Reliable Systems (STARS) program, is approved for release under Distribution "A" of the Scientific and Technical Information Program Classification Scheme (DoD Directive 5230.24) unless otherwise indicated. Sponsored by the U.S. Defense Advanced Research Projects Agency (DARPA) under contract F19628-88-D-0031, the STARS program is supported by the military services, SEI, and MITRE, with the U.S. Air Force as the executive contracting agent.

Permission to use, copy, modify, and comment on this software and its documentation for purposes stated under Distribution "A" and without fee is hereby granted, provided that this notice appears in each whole or partial copy. This software retains Contractor indemnification to The Government regarding copyrights pursuant to the above referenced STARS contract. The Government disclaims all responsibility against liability, including costs and expenses for violation of proprietary rights, or copyrights arising out of the creation or use of this software.

In addition, the Government, Paramax, and its subcontractors disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness, and in no event shall the Government, Paramax, or its subcontractor(s) be liable for any special, indirect or consequential damages or any damages whatsoever resulting from the loss of use, data, or profits, whether in action of contract, negligence or other tortious action, arising in connection with the use or performance of this software.

TASK: UU04
CDRL: 05203
24 February 1993

VERSION DESCRIPTION DOCUMENT
Ada PCTE Binding (AdaPCTE)
Version 0.3
SunOS Implementation

Principal Author(s):

Robert C. Smith, Paramax, Valley Forge Engineering Center *Date*

Michael J. Horton, Paramax, Valley Forge Engineering Center *Date*

Approvals:

Paul J Orgren 5/10/93

Task Manager *Dr. Paul Orgren* *Date*

(Signatures on File)

TASK: UU04
CDRL: 05203
24 February 1993

VERSION DESCRIPTION DOCUMENT
Ada PCTE Binding (AdaPCTE)
Version 0.3
SunOS Implementation

Change Record:

Data ID	*Description of Change*	*Date*	*Approval*
STARS-UC-05203/004/00	Successor Volume: Upgrade for software version 0.3	24 February 1993	*on file*
STARS-UC-05203/003/00	Successor Volume: Upgrade for software version 0.2	30 November 1992	*on file*
STARS-TC-04014/001/00	Original Issue	12 June 1992	*on file*

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	24 February 1993	Informal Technical Report

4. TITLE AND SUBTITLE

Ada PCTE Binding (AdaPCTE) Version 0.3
SunOS Implementation

5. FUNDING NUMBERS

F19628-88-D-0013

6. AUTHOR(S)

Bob Smith
Michael Horton

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Paramax Systems Corporation
12010 Sunrise Valley Drive
Reston, VA 22091

8. PERFORMING ORGANIZATION REPORT NUMBER

STARS-UC-05203/004/00

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Department of the Air Force
Headquarters ESC
Hanscom, AFB, MA 01731-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

05203

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution "A"

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The Ada Portable Common Tool Environment (PCTE) binding (AdaPCTE) provides Ada applications access to a PCTE object base as defined by the European Computer Manufacturers Association (ECMA) Ada PCTE specification (Standard ECMA-162 Ada Language Binding, December 1991). This "alpha" release provides a minimal set of interfaces to permit Ada developers to experiment with and evaluate PCTE for use in future Ada applications. It is expected that later releases will complete the binding.

14. SUBJECT TERMS

Ada PCTE Bindings

15. NUMBER OF PAGES

16

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

Contents

1 SCOPE **1**

1.1 Identification 1

1.2 System Overview 1

2 RELATED SOFTWARE **1**

3 VERSION DESCRIPTION **1**

3.1 Inventory of Contents 1

Δ 3.1.1 Directory: *adapcte/code* 1

3.1.2 Sub-directory: *adapcte/code/C* 2

3.2 Changes Installed 2

Δ 3.2.1 Version 0.3 2

3.2.2 Version 0.2 3

3.3 Adaptation Data 3

3.3.1 Operating Environment 3

3.3.2 Development Environment 3

3.4 Interface Compatibility 4

3.5 Installation Instructions 4

3.5.1 Build Procedure 4

4 USER FEEDBACK **4**

5 NOTES **5**

A Appendix: Inventory of Contents **8**

B Appendix: Build Scripts **10**

B.1 Script: *Build_AdaPCTE.var* 10

B.2 Script: *Build_AdaPCTE.csh* 14

1 SCOPE

1.1 Identification

Version Description Document,
Ada PCTE Binding (AdaPCTE),
Δ Version 0.3,
SunOS Implementation

1.2 System Overview

The Ada Portable Common Tool Environment (PCTE) binding (AdaPCTE) provides Ada applications access to a PCTE object base as defined by the European Computer Manufacturers Association (ECMA) Ada PCTE specification (Standard ECMA-162 Ada Language Binding, December 1991). This "alpha" release provides a minimal set of interfaces to permit Ada developers to experiment with and evaluate PCTE for use in future Ada applications. It is expected that later releases will complete the binding.

2 RELATED SOFTWARE

Since no conforming implementations of ECMA PCTE exist as defined in Standard ECMA-149, AdaPCTE is implemented on GIE Emeraude's PCTE V12.2 Fix 7 or V12.3. Because
Δ only a subset of the ECMA PCTE Ada specification has been implemented for the 0.3 release, and because ECMA PCTE functionality differs somewhat from Emeraude PCTE functionality, the complete functionality of Emeraude PCTE is not available to Ada applications using these bindings.

3 VERSION DESCRIPTION

3.1 Inventory of Contents

The AdaPCTE distribution is structured as shown below. The top-level directory *adapcte* includes PostScript (**VDDadapcte.ps**) and clear ASCII text (**VDDadapcte.tty**) versions of this document, along with a complete directory listing of the distribution (**Contents.tty**, reproduced herein as **Appendix A**).

Δ 3.1.1 Directory: *adapcte/code*

The *adapcte/code* directory contains the Ada source files for the Ada binding to PCTE and the UNIX C-shell script *Build_AdaPCTE.csh*. *Build_AdaPCTE.csh* can be used to build
Δ the entire AdaPCTE Binding using the SunAda 1.1 Development System. No provisions within the build script have been made for installing the bindings in the PCTE object base.

Applications being developed on these bindings are expected to be developed within a UNIX environment and executable code files may be installed by the user in the PCTE object base (but are not required to be installed in the object base). The build script is reproduced herein as **Appendix B.2**.

3.1.2 Sub-directory: *adapcte/code/C*

This directory contains a small C file, *util.c*, containing utility routines used by AdaPCTE. The build script compiles this file in the target directory as *util.o*. Applications are required to add this file to any link commands for applications developed with these bindings.

3.2 Changes Installed

Δ 3.2.1 Version 0.3

In the original ECMA-162 Ada Language Binding specification for the file *Pcte.a* a generic package specification for sequences was declared and instantiated before the package body was elaborated. This violates MIL-STD-1815A-1983 (the Ada Reference Manual). The SunAda 1.0 compiler did not enforce generic package elaboration before instantiation. The SunAda 1.1 compiler now enforces this, so to successfully compile and execute AdaPCTE-based applications, the generic sequences package and it's body were moved to a separate compilation unit. This caused several minor changes to other AdaPCTE files, and may require changes to applications using the AdaPCTE bindings. Application changes should be limited to changing data types of paramaters passed to procedures in the sequences package.

The ECMA standards group responsible for the ECMA-162 has been notified of the problem, and a description of changes made to AdaPCTE provided. Future releases of the standard may differ from the specifications in AdaPCTE 0.3. ECMA changes will be incorporated in future releases of AdaPCTE.

Files affected were:

```
Pcte.a
Pcte_accounting.a
Pcte_audit.a
Pcte_b.a
Pcte_discretionary.a
Pcte_oms.a
Pcte_process.a
Pcte_process_b.a
Sequences.a
Sequences_b.a
```

Several inconsistencies between standard Ada text input-output and PCTE text input-output were discovered. Changes were made to *pcte_text_io* so that *get* and *end_of_line* work as in standard Ada text input-output. Files affected were:

```
Pcte_contents_b.a
Pcte_text_io_b.a
```

3.2.2 Version 0.2

Added to this version are Ada Input Output packages that support text, sequential, and direct input output to PCTE objects with contents. The specifications of these packages match the standard Ada input output routines defined in the Language Reference Manual. The package bodies have been rewritten to access PCTE objects. The packages are specified in *pcte_text_io.a*, *pcte_direct_io.a*, and *pcte_sequential_io.a*.

To support access to Emeraude PCTE objects a package *pcte_support* was added. This package provides constants for accessing standard PCTE objects (e.g., common root, home object), and routines for manipulating Emeraude PCTE path names.

The procedure *list_all_links* in package *Pcte_object* was extended to permit applications to request a listing of links based on an application supplied list of link types.

3.3 Adaptation Data

3.3.1 Operating Environment

Sun-4 Workstations

SunOS, Version 4.1.2

Emeraude PCTE V12.2 Fix 7 or V12.3

3.3.2 Development Environment

Sun-4 Workstations (32 MB main memory, 100 MB swap space)

SunOS, Version 4.1.2

Emeraude PCTE V12.2 Fix 7 or V12.3

Δ SunAda 1.1

C compiler

3.4 Interface Compatibility

AdaPCTE uses the recently adopted standard ECMA-162 for the Ada binding specification. Because no ECMA PCTE implementation is available, AdaPCTE is bound to GIE Emeraude's PCTE 1.5 V12.2 or V12.3 implementation written in "C". As a result, the AdaPCTE specification contains some minor modifications to ECMA-162. The exact specification of the implemented binding can be found in the Ada package specifications located in the directory */adapcte/code* in this delivery.

3.5 Installation Instructions

File *adapcte/code/Build_AdaPCTE.csh* is an executable UNIX C-shell script, which can be used interactively to build the AdaPCTE Binding from the Ada source code, using the SunAda 1.0 system. It ensures that library dependencies are established correctly, making it unnecessary for the installer to perform these operations manually.

3.5.1 Build Procedure

1. *(OPTIONAL)* - To prevent interactive prompting when executing the script, uncomment and edit the environment variables at the beginning of file *code/Build_AdaPCTE.var* (see **Appendix B.1**) to reflect the actual operating environment. The following environment variables must be modified:

   AdaPCTE - identifies the full pathname of the directory into which the AdaPCTE distribution has been loaded (e.g., /local/adapcte);

   COMPILERNAME - identifies the name of the compiler to be used;

   COMPVERSION - identifies the compiler version;

   COMPILERPATH - identifies the full pathname of the directory containing the SunAda compilation system (e.g., /local/sunada1.0);

   TARGET - identifies a *Build* directory to be used for building the software.

2. Execute *Build_AdaPCTE.csh*, providing configuration information when prompted by the script.

4 USER FEEDBACK

This version of AdaPCTE is considered an "alpha" release. The primary purpose of the release is to encourage experimentation with the software and to solicit feedback from the Ada and PCTE user communities. Thus, we would greatly appreciate your comments, suggestions, and criticisms.

5 NOTES

The full set of PCTE path names as described in the ECMA PCTE Abstract Specification (149) has not been implemented for this release. The following characters "_", ".", "~", and "/" plus alphanumeric characters are valid characters in AdaPCTE path names. The following are examples of valid AdaPCTE path names:

```
_/sun4.tools
~/.history.e
```

Δ AdaPCTE Version 0.3 has not implemented all the interfaces defined in ECMA-162. The following describes which interfaces are implemented in Version 0.3 including any limitations.

```
Package Sequence
   function get
   procedure put
   procedure delete
   procedure copy
   function length_of
   function index_of
   function equal
   procedure normalize

Package Pcte
   Package Reference
         -- These procedures use a limited form of path names as defined
         -- in the abstract spec.  You can use ~, _, ., .., / plus ascii
         -- characters
      function get_path
      procedure set_absolute
      procedure set_relative
      procedure unset
      -- New operations added by VFEC
            function get_reference_id
            procedure set_reference_id

Package Pcte_contents
      --  This package is only implemented for files; no pipes or devices
   procedure close
   function get_position
   procedure open
   function read
   procedure seek
```

```
   procedure set_position
   procedure set_properties
   procedure write
   -- New operations added by VFEC
      procedure standard_input
      procedure standard_output
      procedure standard_error
      function end_of_contents
      procedure write_s
         -- (writes a string)
      procedure read_s
          -- (reads a string)

Package Pcte_error
   procedure set
   procedure unset
   procedure set_will_raise
   procedure set_will_record
   function will_raise
   function will_record
   function last_error

Package Pcte_object
   procedure create
      -- can not specify another volume
   procedure delete
   procedure get_attribute
      -- for boolean, integer, natural and string types only
   procedure get_several_attributes
      -- for boolean, integer, natural and string types only
   function get_type
   procedure list_all_links
      -- does not support EXTERNAL extents
      -- does not support COMPOSITE scopes
      -- ignores links parameter
      -- none of the other 8 procedure variations of
      -- object_list_links is supported

Package Pcte_process
   procedure create_and_start
      -- no process objects created; just fire up a process
      -- local execution site only
   procedure set_working_schema
      -- for current process only
   procedure wait_for_any_child
```

```
   procedure wait_for_child

package Pcte_sds
   procedure get_link_type_properties
   procedure get_object_type_properties
   function get_type_name
      -- ignores any sds param value other than IN_WORKING_SCHEMA

package Pcte_link
   procedure get_attribute
      -- for boolean, integer, natural and string types only
   procedure get_several_attributes
      -- for boolean, integer, natural and string types only
```

A Appendix: Inventory of Contents

```
NOTE:  "*" identifies executables; "/" identifies directories

adapcte:
Contents.tty
VDDadapcte.ps
VDDadapcte.tty
code/

adapcte/code:
Build_AdaPCTE.csh*
Build_AdaPCTE.var
C/
Pcte.a
Pcte_accounting.a
Pcte_activity.a
Pcte_audit.a
Pcte_b.a
Pcte_contents.a
Pcte_contents_b.a
Pcte_discretionary.a
Pcte_discretionary_b.a
Pcte_error.a
Pcte_error_b.a
Pcte_limit.a
Pcte_mandatory.a
Pcte_mandatory_b.a
Pcte_message.a
Pcte_notify.a
Pcte_object_b.a
Pcte_oms.a
Pcte_oms_b.a
Pcte_process.a
Pcte_process_b.a
Pcte_queue.a
Pcte_replicated_object.a
Pcte_sds.a
Pcte_sds_b.a
Pcte_time.a
Pcte_vol_dev_archi.a
Pcte_workstation.a
Sequences.a
Sequences_b.a
emer_conversion.a
```

```
error.a
error_b.a
errors_c.a
pcte_1_5_int.a
pcte_1_5_support.a
pcte_1_5_support_b.a
pcte_direct_io.a
pcte_direct_io_b.a
pcte_object_create.a
pcte_sequential_io.a
pcte_sequential_io_b.a
pcte_support.a
pcte_support_b.a
pcte_text_io.a
pcte_text_io_b.a

adapcte/code/C:
util.c
```

B Appendix: Build Scripts

B.1 Script: *Build_AdaPCTE.var*

```
1  #
2  #  Uncomment and edit these lines if you do not want to
3  #  be prompted for the environment variables
4  #
5  setenv ADAPCTE       /tbd/adapcte     # replace with location of the release
6  setenv COMPILERNAME  sunada           # set to sunada
7  setenv COMPVERSION   SunAda1.1        # e.g. SunAda1.0 or SunAda1.1
8  setenv COMPILERPATH  /tbd/SunAda1.1   # indicate your compiler path
9  setenv TARGET        $ADAPCTE/Build_$COMPVERSION
10
11  #
12  # Define the location of the  RGB source code directories.
13  #
14
15  if ( ! $?ADAPCTE ) then
16    echo ""
17    echo "Specify path to top level Ada PCTE directory "
18    echo "(e.g. /local/adapcte ) "
19    echo ""
20    echo -n "          ADAPCTE = "
21    setenv ADAPCTE $<
22    echo ""
23  endif
24  if ( ! -e $ADAPCTE ) then
25    echo ""
26    echo "** $ADAPCTE does not exist **"
27    echo "** Script aborted **"
28    echo ""
29    unsetenv ADAPCTE
30    exit -1
31  endif
32
33
34
35  #
36  # Define C Language compilation variable
37  #
38  setenv CC            " cc -g -c "
39
40
41
```

```
#
# Determine the Ada compilation system to use
#
#
# Establish a path to the SunAda compilation system
#
if ( ! $?COMPILERNAME || ! $?COMPVERSION || ! $?COMPILERPATH ) then
  echo ""
  echo "Please select your compiler name:  [sunada] "
  echo ""
  echo -n "  COMPILERNAME = "
  setenv COMPILERNAME $<
  echo ""
  switch ($COMPILERNAME)
    case Vads:
    case VADS:
    case vads:
      echo -n "Are you building with VADS Version 6.0.3? [y,n](n) "
      set COMPVERSION = $<
      echo ""
      switch ($COMPVERSION)
        case Y:
        case y:
          set COMPVERSION = Vads603
          breaksw
        case N:
        case n:
        default:
          set COMPVERSION = Vads
          echo ""
          echo "Warning!  Software not tested under your version of the VADS com
piler."
        breaksw
      endsw
    breaksw
    case SunAda:
    case Sunada:
    case sunada:
      echo -n "Which version of SunAda are you using? [0,1](0) "
      set COMPVERSION = $<
      echo ""
      switch ($COMPVERSION)
        case 1:
          set COMPVERSION = SunAda1.1
          echo "Warning!  Software not tested under your version of the SunAda compiler."
```

```
87            breaksw
88          case 0:
89          default:
90            set COMPVERSION = SunAda1.0
91          breaksw
92        endsw
93      breaksw
94      default:
95        echo ""
96        echo "You must specify a compiler name."
97        echo ""
98        unsetenv COMPVERSION
99        exit -1
100        breaksw
101     endsw
102
103     echo ""
104     echo "Specify path to the compiler (e.g. /local/SunAda)"
105     echo ""
106     echo -n "   COMPILERPATH = "
107     setenv COMPILERPATH $<
108     if ( ( $COMPILERPATH == ) || ( ! -e $COMPILERPATH/bin/ada ) ) then
109       echo ""
110       echo "** Cannot find Ada compiler in $COMPILERPATH/bin **"
111       echo "** Script aborted **"
112       echo ""
113       unsetenv COMPILERPATH
114       exit -1
115     endif
116 endif
117 if ( -e $COMPILERPATH/bin/ada ) then
118   if ( $COMPILERNAME == "sunada" || $COMPILERNAME == "vads" ) then
119     setenv COMPILERBIN $COMPILERPATH/bin
120     setenv COMPILE     "$COMPILERBIN/ada -v -O0 "
121     setenv LINK        "$COMPILERBIN/a.ld "
122   endif
123 else
124   echo ""
125   echo "** Cannot find $COMPILERPATH/bin/ada **"
126   echo "** Script aborted **"
127   echo ""
128   unsetenv COMPILERPATH
129   exit -1
130 endif
131
```

```
132
133  #
134  # Define the Destination of the ADAPCTE build
135  #   where  TARGET = path to build destination (e.g. $ADAPCTE/Build_SunAda1.0)
136  #
137  if ( ! $?TARGET ) then
138    echo ""
139    echo "Specify the path to the TARGET directory "
140    echo "(Defaults to $ADAPCTE/Build_${COMPVERSION}) "
141    echo ""
142    echo -n "         TARGET = "
143    setenv TEMP $<
144    echo ""
145    if ( $TEMP == ) then   # check for null entry
146      setenv TARGET  $ADAPCTE/Build_${COMPVERSION}
147      unsetenv TEMP
148    else
149      setenv TARGET $TEMP
150      unsetenv TEMP
151    endif
152  endif
153
154  echo ""
155  echo "          TARGET = $TARGET"
156  echo ""
157  echo "        ADAPCTE = $ADAPCTE"
158  echo ""
159  echo "   COMPILERNAME = $COMPILERNAME"
160  echo "    COMPVERSION = $COMPVERSION"
161  echo "   COMPILERPATH = $COMPILERPATH"
162  echo "        COMPILE = $COMPILE"
163  echo "           LINK = $LINK"
164
165  #
166  # Create the directories for the build
167  #
168  if ( ! -d $TARGET ) mkdir $TARGET
169
170
171
```

B.2 Script: *Build_AdaPCTE.csh*

```
1  #! /bin/csh -f
2  echo ""
3  echo "Defining installation-dependent variables"
4  echo ""
5  source Build_AdaPCTE.var
6
7  echo ""
8  echo "Building Ada libraries for the Ada Bindings to PCTE"
9  echo ""
10
11 if ! -e $TARGET mkdir $TARGET
12
13 cd $TARGET
14
15 if ( ( $COMPILERNAME == "vads" ) || ($COMPILERNAME == "sunada" ) ) then
16   if ( ! -e ada.lib ) then
17     $COMPILERBIN/a.mklib -f $TARGET $COMPILERPATH/verdixlib
18   endif
19 endif
20
21 echo ""
22 echo "Creating source code links in $ADAPCTE/code"
23 echo ""
24 foreach file ($ADAPCTE/code/*.a)
25   if (! -e ${file:t} ) ln -s $file ${file:t}
26 end
27
28 foreach file ($ADAPCTE/code/C/*)
29   if (! -e ${file:t} ) ln -s $file ${file:t}
30 end
31
32 rm -rf LOGadapcte
33
34 echo ""
35 echo "Compiling the Ada PCTE binding source"
36 echo ""
37
38 $COMPILE Sequences.a                    >>& LOGadapcte
39 $COMPILE Sequences_b.a                  >>& LOGadapcte
40 $COMPILE Pcte_error.a                   >>& LOGadapcte
41 $COMPILE Pcte.a                         >>& LOGadapcte
42 $COMPILE Pcte_contents.a                >>& LOGadapcte
43 $COMPILE Pcte_replicated_object.a       >>& LOGadapcte
```

```
44  $COMPILE Pcte_message.a                >>& LOGadapcte
45  $COMPILE Pcte_error_b.a                >>& LOGadapcte
46  $COMPILE Pcte_notify.a                 >>& LOGadapcte
47  $COMPILE Pcte_discretionary.a          >>& LOGadapcte
48  $COMPILE Pcte_mandatory.a              >>& LOGadapcte
49  $COMPILE Pcte_audit.a                  >>& LOGadapcte
50  $COMPILE Pcte_mandatory_b.a            >>& LOGadapcte
51  $COMPILE Pcte_workstation.a            >>& LOGadapcte
52  $COMPILE Pcte_discretionary_b.a        >>& LOGadapcte
53  $COMPILE Pcte_process.a                >>& LOGadapcte
54  $COMPILE emer_conversion.a             >>& LOGadapcte
55  $COMPILE Pcte_vol_dev_archi.a          >>& LOGadapcte
56  $COMPILE errors_c.a                    >>& LOGadapcte
57  $COMPILE error.a                       >>& LOGadapcte
58  $COMPILE error_b.a                     >>& LOGadapcte
59  $COMPILE pcte_1_5_int.a                >>& LOGadapcte
60  $COMPILE pcte_1_5_support.a            >>& LOGadapcte
61  $COMPILE pcte_1_5_support_b.a          >>& LOGadapcte
62  $COMPILE Pcte_process_b.a              >>& LOGadapcte
63  $COMPILE Pcte_contents_b.a             >>& LOGadapcte
64  $COMPILE Pcte_b.a                      >>& LOGadapcte
65  $COMPILE Pcte_oms.a                    >>& LOGadapcte
66  $COMPILE Pcte_object_b.a               >>& LOGadapcte
67  $COMPILE Pcte_oms_b.a                  >>& LOGadapcte
68  $COMPILE Pcte_time.a                   >>& LOGadapcte
69  $COMPILE Pcte_sds.a                    >>& LOGadapcte
70  $COMPILE Pcte_sds_b.a                  >>& LOGadapcte
71  $COMPILE Pcte_queue.a                  >>& LOGadapcte
72  $COMPILE Pcte_accounting.a             >>& LOGadapcte
73  $COMPILE Pcte_activity.a               >>& LOGadapcte
74  $COMPILE Pcte_limit.a                  >>& LOGadapcte
75  $COMPILE pcte_support.a                >>& LOGadapcte
76  $COMPILE pcte_support_b.a              >>& LOGadapcte
77  $COMPILE pcte_object_create.a          >>& LOGadapcte
78  $COMPILE pcte_text_io.a                >>& LOGadapcte
79  $COMPILE pcte_text_io_b.a              >>& LOGadapcte
80  $COMPILE pcte_direct_io.a              >>& LOGadapcte
81  $COMPILE pcte_direct_io_b.a            >>& LOGadapcte
82  $COMPILE pcte_sequential_io.a          >>& LOGadapcte
83  $COMPILE pcte_sequential_io_b.a        >>& LOGadapcte
84
85
86  echo ""
87  echo "Compiling the C code"
88  echo ""
```

```
89  $CC util.c                          >>&  LOGadapcte
90
91  echo ""
92  echo "Compilation Complete"
```